

This listing of claims will replace all prior versions, and listings, of claims in the application:

LISTING OF THE CLAIMS:

1. (Currently Amended) ~~A~~ The message processing apparatus characterized by having
according to claim 6, further comprising:

~~process requestor search information management means for managing process~~
~~requestor search information for searching for an applicable process requestor as to an agent~~
~~start cause event;~~

~~acceptance means for accepting the agent start cause event;~~

~~list information creation means for, based on said process requestor search~~
~~information, creating list information on process requestors to which a message generated due~~
~~to said agent start cause event~~ agent activating event is applied;

~~a plurality of agents associated with the process requestors, stored in a persistent~~
~~storage, readable from the persistent storage to a cache memory as a program execution area~~
~~and discardable from the cache memory, each agent operating only when existing in the cache~~
~~memory to be able to process the message in a message queue corresponding to the agent;~~

~~insertion and reading means for, of the process requestors included in said list~~
~~information, selecting a plurality of unselected ones~~ process requestors ~~as the process~~
~~requestors to be inserted and read, inserting said message into the message queues related to~~
~~the process requestors to be inserted and read and reading the agents related to said process~~
~~requestors from the persistent storage to the cache memory;~~

~~agent instruction means for instructing the agent related to the message queue having the message inserted therein to operate; and~~

~~repetitive instruction means for, in the case where the unselected one process requester remains among the process requesters included in said list information, waiting for termination of the process of all the agents in operation and instructing said insertion and reading means to repeat the process.~~

2. (Currently Amended) The message processing apparatus according to claim 1, ~~characterized in that~~ wherein each agent sends a notice to the process requester associated with the agent as to the message in the message queue associated with the agent.

3. (Currently Amended) A The message processing apparatus ~~characterized by having~~ according to claim 6, further comprising:

~~process requester search information management means for managing process requester search information for searching for an applicable process requester as to an agent start cause event;~~

~~acceptance means for accepting the agent start cause event;~~

~~list information creation means for, based on said process requester search information, creating list information on process requesters to which a message generated due to said agent start cause event agent activating event is applied;~~

~~a plurality of agents associated with the process requesters, stored in a persistent storage, readable from the persistent storage to a cache memory as a program execution area and abandonable from the cache memory, each agent operating only when existing in the~~

~~cache memory to be able to process the message in a message queue corresponding to the agent;~~

a first message queue processing mechanism;

a second message queue processing mechanism; and

selection means for selecting either one of the first and second message queue processing mechanisms; and,

~~agent instruction means for, as to the agents related to the message queue having the message inserted therein, immediately instructing the agent to operate if the agent is in the cache memory, and reading the agent from said persistent storage to said cache memory and then instructing the agent to operate if the agent is not in the cache memory; and wherein:~~

the first message queue processing mechanism has insertion means for inserting said message into the message queues related to all the process requesters included in said list information; and

the second message queue processing mechanism has:

insertion and reading means for, of the process requesters included in said list information, selecting a plurality of unselected ~~ones~~ process requesters as the process requestors to be inserted and read, inserting said message into the message queues related to the process requestors to be inserted and read and reading the agents related to said process requestors from the persistent storage to the cache memory; and

repetitive instruction means for, in the case where the unselected ~~one~~ process requester remains among the process requestors included in said list information, waiting for termination of the process of all the agents in operation and instructing said insertion and reading means to repeat the process.

4. (Currently Amended) The message processing apparatus according to claim 3,
~~characterized in that~~ wherein selection of said selection means is based on an
instruction of an operator.

5. (Currently Amended) The message processing apparatus according to claim 3,
~~characterized in that~~ wherein selection of said selection means is based on ~~the~~ estimated
number of the process requesters to which the message generated ~~due to~~ upon the agent start
~~cause event~~ agent activating event ~~of this time is applied~~, estimated hit rate in the cache
memory where the agents resides, said agents associated with ~~as to the agents related to the~~
process requesters to which the message generated ~~due to~~ upon the agent start ~~cause event~~
agent activating event ~~of this time is applied~~, estimated work time, in the case where said
selection means selects the first message queue processing mechanism, from acceptance of
~~the~~ information by said acceptance means until obtaining the list information on the process
requestors to which the message is applied and inserting the message into the message queues
of all the agents, estimated work time, in the case where said selection means selects the
second message queue processing mechanism, from acceptance of ~~the~~ information by said
acceptance means until obtaining the list information on the process requesters to which the
message is applied and inserting the message into the message queues of all the agents,
estimated time from determining ~~determination of use as to the agent in~~ of the cache memory
associated with the agent until ~~completion of completing the a process by the a determined~~
agent, and/or estimated time from ~~determination of~~ determining use as to the agent of outside

the cache memory associated with the agent until the ~~completion of~~ completing the process by the determined agent.

6. (Currently Amended) A message processing apparatus ~~characterized by having comprising:~~

process requester search information management means for managing process requestor search information for searching for an applicable process requester associated with ~~as to agent start cause event~~ agent activating event;

acceptance means for accepting the ~~agent start cause event~~ agent activating event;

process requestor determination means for determining ~~the a~~ process requestor to which ~~the a~~ message generated ~~due to~~ upon said ~~agent start cause event~~ agent activating event is ~~applied~~ based on said process requestor search information;

a plurality of agents associated with the process requesters, stored in a persistent storage, readable from the persistent storage to a cache memory ~~as a program execution area~~ and abandonable from the cache memory, each agent becoming operable when existing in the cache memory;

at least one sub-process priority determination means for determining process priority about each message as sub-process priority based on a single standard of value;

compound process priority determination means for, when the total number of said sub-process priority determination means is two or more, determining compound process priority about each message based on the sub-process priority individually determined as to each message by each sub-process priority determination means, and when the total number of said sub-process priority determination means is one, determining as the compound process

priority the sub-process priority determined by said one sub-process priority determination means as to each message; and

agent instruction means for rendering the message of the highest compound process priority among the messages held by each message queue as the message of the highest priority, and between the agents related to the message queues of which compound process priority of the message of the highest priority is the same, instructing the agent existing in the cache memory to operate in preference to the agent not existing therein.

7. (Currently Amended) The message processing apparatus according to claim 6, ~~characterized in that~~ wherein said standard of value may be related to the contents of the message or to the process requesters to which the message is applied.

8. (Currently Amended) The message processing apparatus according to claim 7, ~~characterized in that~~ wherein the predetermined standard of value related to the contents of the message includes the standard of value related to emergency of ~~processing of the message~~.

9. (Currently Amended) The message processing apparatus according to claim 7, ~~characterized in that~~ wherein the standard of value related to the process requestors to which the message is applied includes the standard of value related to rating of the process requestors.

10. (Currently Amended) The message processing apparatus according to claim 6,

~~characterized in that~~ wherein, when there are a plurality of messages held by the message queue ~~related to~~ associated with the agent ~~initiated~~ which once started the process by said agent instruction means, the agent continuously processes all those messages or the messages of which compound process priority is within a predetermined ~~position~~ value in descending rank.

11. (Currently Amended) The message processing apparatus according to claim 6, ~~characterized in that~~ wherein:

there is agent management means including said sub-process priority determination means and said compound process priority determination means;

said agent management means has existence detection means for detecting whether or not each agent exists in the persistent storage, grouping information management means for grouping the agents and managing grouping information based on ~~determination~~ results of the existence detection means and the compound process priority of each agent, and update instruction means for instructing the grouping information management means to update the grouping information; and

said agent instruction means instructs the agents to operate in order based on the grouping information of said agent management means.

12. (Currently Amended) ~~A~~ The message processing apparatus characterized by having according to claim 6, further comprising:

~~process requester search information management means for managing process requester search information for searching for an applicable process requester as to an agent start cause event;~~

~~acceptance means for accepting the agent start cause event;~~

acceptance order information management means for managing acceptance order information on the ~~agent start cause events~~ agent activating events accepted by said acceptance means;

~~a plurality of agents associated with the process requestors, stored in a persistent storage, readable from the persistent storage to a cache memory as a program execution area and abandonable from the cache memory, each agent operating only when existing in the cache memory to be able to process the message in a message queue corresponding to the agent;~~

a plurality of threads mutually operable in parallel, each thread detecting the process requestors to which the message generated due to said ~~agent start cause event~~ agent activating events is applied based on said process requester search information and inserting said message into the message queues related to the process requestors;

allocation means for allocating to each thread the ~~agent start cause event~~ agent activating event to be processed by ~~that~~ the thread;

proceeding information management means for managing proceeding information ~~on the process by~~ of the thread as to ~~associated with each agent start cause event~~ agent activating event accepted by said acceptance means;

determination means for, as to ~~associated with the agent start cause event~~ agent activating event of which ~~process~~ indicates proceeding information is the information on

thread ~~process-termination~~ (hereafter, referred to as a "determined agent start cause event agent activating event"), determining whether or not, of the ~~agent start cause events~~ agent activating events accepted by said acceptance means prior to the determined ~~agent start cause event~~ agent activating event, there is any ~~agent start cause event~~ agent activating event of which indicates a thread process is unfinished; and

agent control means for controlling ~~the process by the agent as to~~ associated with the message generated ~~due to~~ upon the determined ~~agent start cause event~~ agent activating event determined as "yes" by said ~~determination means~~ that there is an agent activating event which indicates a thread is unfinished

13. (Currently Amended) The message processing apparatus according to claim 12, ~~characterized by having~~ wherein said agent control means for allowing ~~the process by the agent as to~~ associated with the message generated ~~due to~~ upon the determined ~~agent start cause event~~ agent activating event determined as "no" by said ~~determination means~~ that there is no agent activating event which indicates a thread is unfinished.

14. (Currently Amended) The message processing apparatus according to claim 13, ~~characterized by having~~ wherein said determination means for, in the case where the ~~agent start cause event~~ agent activating event immediately following the ~~agent start cause event~~ agent activating event determined as "no" that there is no agent activating event which indicates a thread is unfinished, in acceptance order is already determined as "yes," that there is an agent activating event which indicates a thread is unfinished, changing the a

determination result from "yes" that there is an agent activating event which indicates a thread is unfinished to "no." that there is no agent activating event which indicates a thread is unfinished.

15. (Currently Amended) The message processing apparatus according to claim 14, ~~characterized by having~~ wherein said agent for, in the case of processing the message queue in which the messages generated due to a plurality of ~~agent start cause events~~ agent activating events of which determination result by said determination means is that there is no agent activating event which indicates a thread is unfinished "no" ~~continue in an acceptance order direction,~~ continuously processing the plurality of continuous messages.

16. (Currently Amended) ~~A~~ The message processing method ~~characterized by having~~ according to claim 21, further comprising:

~~a process requestor search information management step of managing process requestor search information for searching for an applicable process requestor as to an agent start cause event;~~

~~an acceptance step of accepting the agent start cause event;~~

a list information creation step of, based on said process requestor search information, creating list information on process requestors to which a message generated due to said ~~agent start cause event~~ agent activating event is applied;

~~an agent setting step of setting a plurality of agents, the agents being associated with the process requestors, stored in a persistent storage, readable from the persistent storage to a cache memory as a program execution area and abandonable from the cache memory, each~~

~~agent operating only when existing in the cache memory to be able to process the message in a message queue corresponding to the agent;~~

an insertion and reading step of, of the process requestors included in said list information, selecting a plurality of unselected ~~ones~~ process requesters as the process requestors to be inserted and read, inserting said message into the message queues related to the process requestors to be inserted and read, and reading the agents related to said process requestors from the persistent storage to the cache memory;

~~an agent instruction step of instructing the agent related to the message queue having the message inserted therein to operate; and~~

a repetitive instruction step of, in the case where the unselected ~~one~~ process requester remains ~~among the process requestors included~~ in said list information, waiting for termination of ~~the process of~~ all the agents in operation and instructing said insertion and reading step to be repeated.

17. (Currently Amended) The message processing method according to claim 16, ~~characterized in that~~ wherein each agent sends a notice to the process requestor associated with the agent as to the message in the message queue associated with the agent.

18. (Currently Amended) A ~~The~~ message processing method ~~characterized by having~~ according to claim 21, further comprising:

~~a process requestor search information management step of managing process requestor search information for searching for an applicable process requestor as to an agent start cause event;~~

~~an acceptance step of accepting the agent start cause event;~~

a list information creation step of, based on said process requester search information, creating list information on process requestors to which a message generated due to said ~~agent start cause event~~ agent activating event is applied;

~~an agent setting step of setting a plurality of agents, the agents being associated with the process requestors, stored in a persistent storage, readable from the persistent storage to a cache memory as a program execution area and abandonable from the cache memory, each agent operating only when existing in the cache memory to be able to process the message in a message queue corresponding to the agent;~~

a first message queue processing step;

a second message queue processing step; and

a selection step of selecting either one of the first and second message queue processing steps; and

~~an agent instruction step of, as to the agent related to the message queue having the message inserted therein, immediately instructing the agent to operate if the agent is in the cache memory, and reading the agent from said persistent storage to said cache memory and then instructing the agent to operate if the agent is not in the cache memory; and wherein:~~

the first message queue processing step has an insertion step of inserting said message into the message queues related to all the process requestors included in said list information; and

the second message queue processing step has:

an insertion and reading step of, of the process requestors included in said list information, selecting a plurality of unselected ~~ones~~ process requestors as the process requestors to be inserted and read, inserting said message into the message queues related to

the process requesters to be inserted and read and reading the agents related to said process requesters from the persistent storage to the cache memory; and

a repetitive instruction step of, in the case where the unselected ~~one~~ process requester remains ~~among the process requesters included~~ in said list information, waiting for termination of ~~the process of~~ all the agents in operation and instructing said insertion and reading step to be repeated.

19. (Currently Amended) The message processing method according to claim 18, characterized in that wherein the selection in said selection step is based on an instruction of an operator.

20. (Currently Amended) The message processing method according to claim 18, characterized in that wherein the selection in said selection step is based on ~~the estimated~~ number of the process requestors to which the message generated ~~due to~~ upon the agent start ~~cause event~~ agent activating event of this time is applied, estimated hit rate in the cache memory where the agents resides, said agents associated with ~~as to the agents related to~~ the process requesters to which the message generated ~~due to~~ upon the agent start ~~cause event~~ agent activating event of this time is applied, estimated work time, in the case where the process is allocated in the first message queue processing step, from acceptance of the information in said acceptance step until obtaining the list information on the process requestors to which the message is applied and inserting the message into the message queues of all the agents, estimated work time, in the case where the process is allocated in the second

message queue processing step, from acceptance of the information in said acceptance step until obtaining the list information on the process requesters to which the message is applied and inserting the message into the message queues of all the agents, estimated time from determining determination of use as to the agent in of the cache memory associated with the agent until completion of the a process by the a determined agent, and/or estimated time from determination of determining use as to the agent of outside the cache memory associated with the agent until the completion of the process by the determined agent.

21. (Currently Amended) A message processing method ~~characterized by having comprising:~~

a process requester search information management step of managing process requestor search information for searching for an applicable process requester associated with as to agent start cause event agent activating event;

an acceptance step of accepting the ~~agent start cause event~~ agent activating event;

a process requestor determination step of determining the a process requestor to which the a message generated due to upon said agent start cause event agent activating event is applied based on said process requestor search information;

an agent setting step of setting a plurality of agents, the agents being associated with the process requesters, stored in a persistent storage, each agent being readable from the persistent storage to a cache memory ~~as a program execution area~~ and abandonable from the cache memory, and each agent operating only when existing in the cache memory to be able to process the message in a message queue corresponding to the agent;

at least one sub-process priority determination step of determining process priority about each message as sub-process priority based on a single standard of value;

a compound process priority determination step of, when the total number of said sub-process priority determination steps is two or more, determining compound process priority about each message based on the sub-process priority individually determined as to each message in each sub-process priority determination step, and when the total number of said sub-process priority determination steps is one, determining as the compound process priority the sub-process priority determined in said one sub-process priority determination step as to each message; and

an agent instruction step of rendering the message of the highest compound process priority among the messages held by each message queue as the message of the highest priority, and between the agents related to the message queues of which compound process priority of the message of the highest priority is the same, instructing the agent existing in the cache memory to operate in preference to the agent not existing therein.

22. (Currently Amended) The message processing method according to claim 21, ~~characterized in that~~ wherein said standard of value may be related to the contents of the message or to the process requestors to which the message is applied.

23. (Currently Amended) The message processing method according to claim 22, ~~characterized in that~~ wherein the predetermined standard of value related to the contents of the message includes the standard of value related to emergency of ~~processing of the message.~~

24. (Currently Amended) The message processing method according to claim 22, ~~characterized in that~~ wherein the standard of value related to the process requestors to which the message is applied includes the standard of value related to rating of the process requestors.

25. (Currently Amended) The message processing method according to claim 21, ~~characterized in that~~ wherein, when there are a plurality of messages held by the message queue ~~related to~~ associated with the agent initiated ~~which once started the process in said~~ agent instruction step, the agent continuously processes all those messages or the messages of which compound process priority is within a predetermined position value in descending rank.

26. (Currently Amended) The message processing method according to claim 21, ~~characterized in that~~ further comprising:

~~there is~~ an agent management step including said sub-process priority determination step and said compound process priority determination step;

said agent management step has having an existence detection step ~~of~~ for detecting whether or not each agent exists in the persistent storage, and grouping information management step of grouping the agents, managing grouping information and updating the grouping information as appropriate based on ~~determination~~ results of the existence detection step and the compound process priority of each agent; and

said agent instruction step ~~instructs for instructing~~ the agents to operate in order based on the grouping information in said agent management step.

27. (Currently Amended) A ~~The~~ message processing method ~~characterized by having~~ according to claim 21, further comprising:

~~a process requestor search information management step of managing process requestor search information for searching for an applicable process requestor as to an agent start cause event;~~

~~an acceptance step of accepting the agent start cause event;~~

~~an acceptance order information management step of managing acceptance order information on the agent start cause events~~ agent activating events accepted by said acceptance step;

~~an agent setting step of setting a plurality of agents, the agents being associated with the process requestors, stored in a persistent storage, readable from the persistent storage to a cache memory as a program execution area and abandonable from the cache memory, and each agent operating only when existing in the cache memory to be able to process the message in a message queue corresponding to the agent;~~

~~a thread setting step of setting a plurality of threads, the threads being mutually operable in parallel, detecting the process requestors to which the message generated due to the agent start cause event~~ agent activating event ~~is applied~~ based on said process requestor search information and inserting said message into the message queues related to the process requestors;

~~an allocation step of allocating to each thread the agent start cause event~~ agent activating event to be processed by ~~that the~~ thread;

~~a proceeding information management step of managing proceeding information on the process by of each thread as to associated with each agent start cause event~~ agent activating event accepted by said acceptance step;

~~a determination step of, as to associated with the agent start cause event~~ agent activating event ~~of which process indicates~~ proceeding information is the information on thread process-termination (hereafter, referred to as a "determined agent start cause event

agent activating event "), determining whether or not, of the ~~agent start cause events~~ agent activating events accepted in said acceptance step prior to the determined ~~agent start cause event~~ agent activating event, there is any ~~agent start cause event~~ agent activating event of which thread process is unfinished; and

an agent control step of controlling ~~the process by the agent as to~~ associated with the message generated due to upon the determined agent start cause event agent activating event determined as that there is an agent activating event which indicates a thread is unfinished "yes" in said ~~determination step~~.

28. (Currently Amended) The message processing method according to claim 27, ~~characterized by having wherein~~ wherein said agent control step of allowing ~~the process by the agent as to~~ associated with the message generated ~~due to upon~~ the determined ~~agent start cause event~~ agent activating event determined as "no" in said ~~determination step~~ that there is no agent activating event which indicates a thread is unfinished.

29. (Currently Amended) The message processing method according to claim 27, ~~characterized by having wherein~~ wherein said determination step of, in the case where the ~~agent start cause event~~ agent activating event immediately following the ~~agent start cause event~~ agent activating event determined as "no" that there is no agent activating event which indicates a thread is unfinished, in acceptance order is already determined as "yes," that there is an agent activating event which indicates a thread is unfinished, changing the determination result from "yes" that there is an agent activating event which indicates a thread is unfinished to "no," that there is no agent activating event which indicates a thread is unfinished.

30. (Currently Amended) The message processing method according to claim 29, ~~characterized by having wherein~~ wherein said agent setting step of, in the case of causing the agent to process the message queue in which the messages generated due to a plurality of ~~agent start cause events~~ agent activating events of which determination result by said determination step is that there is no agent activating event which indicates a thread is unfinished "no" ~~continue in an acceptance order direction~~, setting said agent to continuously process the plurality of

continuous messages.

31-33. (Canceled)

34. (Original) A computer program product comprising a computer usable medium having computer readable program code means embodied therein for causing message processing, the computer readable program code means in said computer program product comprising computer readable program code means for causing a computer to effect the functions of claim 6.

35-41. (Canceled)

42. (Original) A program storage device readable by machine, tangibly embodying a program of instructions executable by the machine to perform method steps for message processing, said method steps comprising the steps of claim 21.

43. (Canceled)

44. (New) The message processing apparatus according to claim 6, further comprising: an agent server for processing said plurality of agents asynchronously, having said message queue of said each agent, managing said cache memory, and allocating threads to said agents on said cache memory.

45. (New) A message processing method according to claim 21, further comprising: an agent server step of providing an agent server to process said plurality of agents asynchronously, to have said message queue of said each agent, to manage said cache memory, and to allocate threads to said agents on said cache memory.